Exam.Code:1007
Sub. Code: 7375

2123

M.E. (Information Technology)
Third Semester
MEIT-3105: Advanced Algorithm Analysis and Data Structure

Time allowed: 3 Hours **Max. Marks: 50**

NOTE: Attempt five questions in all, including Question No. 1 (Section -A) which is compulsory and selecting two questions each from Section B-C.

x-x-x

Section-A

1. a) Solve the following recurrence relation using Master's theorem- **T(n) = 2T(n/2) + nlogn** 10

b) Briefly explain the key principle behind Greedy Algorithms and mention one scenario where a Greedy approach might not yield an optimal solution.

c) What is a Minimum Spanning Tree? Write brief steps of Kruskal's algorithm.

d) Define N-Queen problem. How do you solve the N-Queen problem?

e) Suppose the numbers 7 , 5 , 1 , 8 , 3 , 6 , 0 , 9 , 4 , 2 are inserted in that order into an initially empty binary search tree. The binary search tree uses the usual ordering on natural numbers. What is the in-order traversal sequence of the resultant tree?

Section-B

2. Consider the following recursive algorithm for computing the factorial of a non-negative integer:

```
def factorial(n):
  if n == 0 or n == 1:
    return 1
  else:
    return n * factorial(n - 1)
```

i) Write down the recurrence relation for the time complexity of the factorial function. 3

ii) Suppose you have an algorithm with a worst-case time complexity of O(n^2), an average-case time complexity of O(n log n), and a best-case time complexity of O(n). Explain the differences among these complexities and under what conditions each complexity is likely to occur. 5

iii) Using Big-O notation, provide the upper bound for the time complexity of the factorial function. 2

3. a) Consider the Quick Sort algorithm. Discuss the basic idea behind Quick Sort and explain how it utilizes the Divide and Conquer strategy. Provide a recurrence relation for the time complexity of Quick Sort when you divide the list in the ratio 1:10 at each step. Solve it to express the overall time complexity. 5

b) Consider the Single Source Shortest Path problem. Explain how Dijkstra's algorithm, finds the shortest paths in a weighted graph. Discuss the conditions under which Dijkstra's algorithm provides correct results and analyze its time complexity. 5

4. a) Define Multistage Graph. Write Algorithm for Multistage Graph. 4

b) The Traveling Salesperson Problem is a classic optimization problem with applications in various fields. Discuss the statement of the Traveling Salesperson Problem and explain how 6

Contd……P/2

	Dynamic Programming can be applied to find an optimal solution. Analyze the time and space complexity of the Dynamic Programming approach for solving the Traveling Salesperson Problem.	
	Section-C	
5.	a) The Hamiltonian Circuit problem is a classic NP-complete problem. Describe the Hamiltonian Circuit problem and explain how it can be approached using a backtracking algorithm. Discuss the conditions under which the backtracking algorithm is likely to be efficient or inefficient for solving the Hamiltonian Circuit problem.	5
	b) What is the difference between recursion and backtracking? Can n queen problem be solved using backtracking? Justify your answer	5
6.	a) Describe the search operation in a B-Tree. Explain how B-Trees maintain balance and why this is crucial for achieving efficient search operations.	5
	b) Walk through the process of inserting a new node into a Red-Black Tree. Explain the necessary rotations and color adjustments that may be performed to maintain the Red-Black Tree properties. Illustrate your answer with a simple example.	5
7.	a) Define pattern matching and explain its significance in computer science. Briefly describe the basic idea behind brute-force pattern matching and discuss one limitation of this approach.	5
	b) Explain the concepts of NP-hard and NP-complete classes in the context of computational complexity. Provide definitions for both terms and discuss the relationship between them. Provide an example of an NP-complete problem.	5

x-x-x